11-07-00 A

# UTILITY PATENT APPLICATION TRANSMITTAL

(Only for new nonprovisional applications under 37 CFR 1.53(b))

| | |
|---|---|
| Attorney Docket No. | P00-3251 |
| First Inventor | Greg Pellegrino and Thomas Grieff |
| Title | SYSTEM, MACHINE, AND METHOD FOR MAINTENANCE OF MIRRORED DATASETS THROUGH SURROGATE WRITES DURING STORAGE-AREA NETWORK PARTIAL CONNECTIVITY EVENTS |
| Express Mail Label No. | EL700672270US |



## APPLICATION ELEMENTS

Assistant Commissioner for Patents
Box Patent Application
Washington, DC 20231

1. ☒ Fee Transmittal Form (submit an original and a duplicate for fee processing)
2. ☐ Applicant claims small entity status. See 37 CFR 1.27
3. ☒ Specification [ total pages __19__ ] (preferred Arrangement set forth below)
   - Descriptive title of the Invention
   - Cross References to Related Applications
   - Statement Regarding Fed sponsored R&D
   - Reference to sequence listing, a table, or a computer program listing appendix
   - Background of the Invention
   - Brief Summary of the Invention
   - Brief Description of the Drawings
   - Detailed Description
   - Claim(s)
   - Abstract of the Disclosure
4. ☒ Drawing(s) [ total sheets 4 ]
5. ☒ Oath or Declaration [ total sheets _1_ ]
   a. ☒ Newly executed (original or copy)
   b. ☐ Copy from prior appl. (37 C.F.R. § 1.63(d))
      i. ☐ DELETION OF INVENTOR(S) Signed statement attached deleting inventor(s) named in prior application, see 37 C.F.R. §§ 1.63(d)(2) and1.33(b).
6. ☐ Application Data Sheet. (See 37 CFR 1.76
7. ☐ CD-ROM or CD-R in duplicate, large table or Computer Program (Appendix)
8. Nucleotide and/or Amino Acid Sequence Submission (if *applicable, all necessary)*
   a. ☐ Computer Readable Form
   b. ☐ Specification Sequence Listing on
      i. ☐ CD-ROM or CD-R (2 copies); or
      ii. ☐ paper
   c. ☐ Statements verifying identity of above copies

### ACCOMPANYING APPLICATION PARTS

9. ☒ Assignment Papers (coversheet/document(s))
10. ☐ 37 CFR. 3.73(b) Statement (*when there is an assignee*) ☒ Power of Attorney
11. ☐ English Translation Document
12. ☐ IDS & Form 1449 ☐ Copies of IDS Citations
13. ☐ Preliminary Amendment
14. ☒ Return Receipt Postcard (MPEP 503)
15. ☐ Certified Copy of Priority Document(s)
16. ☒ Other: Certificate of Mailing by Express Mail

17. **If a CONTINUING APPLICATION, check appropriate box, and supply the requisite information below and in a preliminary amendment, or in an Application Data Sheet under 37 CFR 1.76:**

☐ Continuation ☐ Divisional ☐ Continuation-in-part (CIP) of prior application No.: __/_____

*Prior application information:* *Examiner:* _____ *Group/Art Unit:* _____

FOR CONTINUATION OR DIVISIONAL APPS only The entire disclosure of the prior application, from which an oath or declaration is supplied under Box 5b, is considered a part of the disclosure of the accompanying continuation or divisional application and is hereby incorporated by reference The incorporation can only be relied upon when a portion has been inadvertently omitted form the submitted application parts

## 17. CORRESPONDENCE ADDRESS

☐ Customer Number or Bar Code Label or ☒ Correspondence address below

| | | | | | |
|---|---|---|---|---|---|
| Name | **William J. KUBIDA, Esq.** | | | | |
| | **Hogan & Hartson, LLP** | | | | |
| Address | **1200 17th Street** | | | | |
| | **Suite 1500** | | | | |
| City | **Denver** | State | CO | ZIP | 80202 |
| Country | **US** | Telephone | **(719) 448-5900** | Fax | **(719) 448-5922** |

| | | | |
|---|---|---|---|
| Name (Print/Type) | **Steven K. Barton** | Registration No. | **36,445** |
| (Signature) | Steven K. Barton | Date | 6 Nov 2000 |

# FEE TRANSMITTAL for FY 2000

| Complete if Known | |
|---|---|
| Application Number | ---------- |
| Filing Date | Herewith |
| First Named Inventor | Greg Pellegrino and Thomas Grieff |
| Examiner Name | |
| Group / Art Unit | |
| Attorney Docket No. | P00-3251 |

| TOTAL AMOUNT OF PAYMENT ($) | ($) 830.00 |
|---|---|

## METHOD OF PAYMENT (check one)

1. ☐ The Commissioner is hereby authorized to charge indicated fees and credit any over payments to:

Deposit Account Number: **50-1123**

Deposit Account Name: **Hogan & Hartson L.L.P.**

☒ Charge Any Additional Fee Required Under 37 CFR § § 1.16 and 1.17

☐ Applicant claims small entity status. See 37 CFR 1.27

2. ☒ **Payment Enclosed:**

☒ Check ☐ Money Order ☐ Other

## FEE CALCULATION

### 1. BASIC FILING FEE

| Large Entity Fee ($) | Small Entity Fee ($) | Fee Description | Fee Paid |
|---|---|---|---|
| 710 | 355 | Utility Filing Fee | 710.00 |
| 320 | 160 | Design filing fee | |
| 490 | 245 | Plant filing fee | |
| 710 | 355 | Reissue filing fee | |
| 150 | 75 | Provisional filing fee | |

SUBTOTAL (1) ($)710.00

### 2. EXTRA CLAIM FEES

| | | | Extra Claims | | Fee from below | = | Fee Paid |
|---|---|---|---|---|---|---|---|
| Total Claims | 20 | -20**= | 0 | X | | = | 0 |
| Independent Claims | 4 | -3**= | 1 | X | 80 | = | 80 |
| Multiple Dependent | | | | | | = | 0 |

***or number previously paid, if greater; For Reissues, see below*

| Large Entity Fee Code | Large Entity Fee ($) | Small Entity Fee Code | Small Entity Fee ($) | Fee Description |
|---|---|---|---|---|
| 103 | 18 | 203 | 9 | Claims in excess of 20 |
| 102 | 80 | 202 | 40 | Independent claims in excessof 3 |
| 104 | 270 | 204 | 135 | Multiple dependent claim, if not paid |
| 109 | 80 | 209 | 40 | **Reissue independent claims over original patent |
| 110 | 18 | 210 | 9 | **Reissue claims in excess of 20 and over original patent |

SUBTOTAL (2) ($)80.00

## FEE CALCULATION (continued)

### 3. ADDITIONAL FEES

| Large Entity Fee ($) | Small Entity Fee ($) | Fee Description | Fee Paid |
|---|---|---|---|
| 130 | 65 | Surcharge – late filing fee or oath | |
| 50 | 25 | Surcharge – late provisional filing fee or cover sheet | |
| 130 | 130 | Non-English specification | |
| 2,520 | 2,520 | For filing a request for ex parte reexamination | |
| 920* | 920* | Requesting publication of SIR prior to Examiner action | |
| 1,840* | 1,840* | Requesting publication of SIR after Examiner action | |
| 110 | 55 | Extension for reply within first month | |
| 390 | 195 | Extension for reply within second month | |
| 890 | 445 | Extension for reply within third month | |
| 1,390 | 695 | Extension for reply within fourth month | |
| 1,890 | 945 | Extension for reply within fifth month | |
| 310 | 155 | Notice of Appeal | |
| 310 | 155 | Filing a brief in support of an appeal | |
| 270 | 135 | Request for oral hearing | |
| 1,510 | 1,510 | Petition to institute a public use proceeding | |
| 110 | 55 | Petition to revive – unavoidable | |
| 1,240 | 620 | Petition to revive – unintentional | |
| 1,240 | 620 | Utility issue fee (or reissue) | |
| 440 | 220 | Design issue fee | |
| 600 | 300 | Plant issue fee | |
| 130 | 130 | Petitions to the Commissioner | |
| 50 | 50 | Petitions related to provisional applications | |
| 240 | 240 | Submission of Information Disclosure Stmt | |
| 40 | 40 | Recording each patent assignment per property (times number of properties) | 40 00 |
| 710 | 355 | Filing a submission after final rejection (37 CFR § 1 129(a)) | |
| 710 | 355 | For each additional invention to be examined (37 CFR §1.129(b)) | |
| 710 | 355 | Request for Continued Examination | |
| | | Request for expedited examination of a design application | |
| | | Other fee (specify) | |

*Reduced by Basic Fling Fee Paid **SUBTOTAL (3)** ($)40.00

| SUBMITTED BY | | | | Complete *(if applicable)* | |
|---|---|---|---|---|---|
| Name *(Print/Type)* | **Steven K. Barton** | Registration No (Attorney/Agent) | **36,445** | Telephone | **(719) 448-5900** |
| *Signature* | Steven K. Barton | | | Date | 6 Nov 2000 |

PATENT
Attorney Docket No.68854.0152
Express Mail No.EL700672270US
Client/Matter No.P00-3251

# SYSTEM, MACHINE, AND METHOD FOR MAINTENANCE OF MIRRORED DATASETS THROUGH SURROGATE WRITES DURING STORAGE-AREA NETWORK TRANSIENTS

## FIELD OF THE INVENTION

The invention relates to the field of high-reliability computing on storage area networks. In particular the inventions relates to systems and methods of maintaining mirrored datasets when a storage area network suffers sufficient disruption that a particular copy of a mirrored dataset can not be seen for direct writes by one, but not all, compute nodes of the network.

## BACKGROUND OF THE INVENTION

In the field of high-reliability computing, it is often desirable to maintain redundant data. Redundant data can provide some protection against failure of a storage device. For example, a RAID (Redundant Array of Independent Disks) system can often be configured to keep full duplicate copies of data on separate disk drives. Should a failure occur that affects one, but not both, of these duplicate, or "mirrored" datasets, data will not be lost. Continued operation may also be possible using the surviving dataset. Other configurations are known, for example, in RAID-5 operation data and parity-recovery information may be striped across a number of drives, failure of any one drive will not result in data loss.

Mirrored datasets are not limited to duplicate datasets maintained by RAID systems. For example, it may be desirable to maintain a primary copy of a mirrored dataset at a different geographical location than the secondary copy. Such remotely located mirrored datasets can provide protection against data loss in the event of flood, fire, lightning strike, or other disaster involving the location of one copy of the dataset.

A mirrored dataset ideally has at least two copies of all information written to the dataset. Whenever a write occurs, that write must be made to both copies for full redundancy to be maintained. If only one copy is written then redundancy protection is lost until repairs can be made and the datasets synchronized. Synchronization of datasets can be a time consuming task; it is desirable that need for synchronization be minimized. On the other hand, reading of data from a mirrored dataset can occur from any copy if the dataset is synchronized, or if the data read is known not to have been altered since the last synchronization of the data.

Storage Area Networks (SANs) are characterized as high-speed networks primarily conveying data between storage nodes and compute nodes, often utilizing separate network hardware from that used for general-purpose network functions. Storage nodes are machines that primarily serve storage to other nodes of the network, while compute nodes are typically computers that use storage provided by storage nodes. Compute nodes may, and often do, have additional storage devices directly attached to them.

SANs are often implemented with fibre-channel hardware, which may be of the arbitrated loop or

switched-fabric type. Storage area networks may be operated in a "clustering" environment, where multiple compute nodes have access to at least some common data, the common data may in turn be stored with redundancy. SANs having multiple processors accessing a common database stored with redundancy, are often used for transaction processing systems.

SANs are also known that use non-fibre-channel interconnect hardware.

Most modern computer networks, including fibre-channel storage area networks, are packet oriented. In these networks, data transmitted between machines is divided into chunks of size no greater than a predetermined maximum. Each chunk is packaged with a header and a trailer into a packet for transmission. In Fibre-Channel networks, packets are known as Frames.

A network interface for connection of a machine, to a Fibre Channel fabric is known as an N_port, and a machine attached to a Fibre-Channel network is known as a node. Nodes may be computers, or may be storage devices such as RAID systems. An NL_port is an N-port that supports additional arbitration required so that it may be connected either to a Fibre Channel fabric or to a Fibre Channel Arbitrated Loop. An L_port is a network interface for connection of a machine to a Fibre Channel Arbitrated Loop. Typically, an N_port, NL_port, or L_Port originates or receives data frames. Each port incorporates such hardware and firmware as is required to transmit and receive frames on the network coupled to a processor and at least one memory system. Ports may incorporate a processor and memory of their own, those that don't utilize memory and processor of their node. Received frames are stored

into memory, and transmitted frames are read from memory. Such ports generally do not re-address, switch, or reroute frames.

SANS often have redundant network interconnect. This may be provided to increase performance by providing high bandwidth between the multiple nodes of the network; to provide for operation despite some potential failures of network components such as hubs, switches, links, or ports; or both.

## DESCRIPTION OF THE PROBLEM

It is possible for some network interconnect components of a SAN to fail while other components continue to operate. This can disrupt some paths between nodes of the network.

There are possible network configurations where a first compute node of the SAN can lose its direct path to a first storage node; while the first compute node has a path to a second storage node of the network, and a second compute node still has a path to the first storage node. If data is mirrored on the primary and secondary storage nodes, the first processor has difficulty updating the copy on the primary storage node, although it can read data from the copy on the secondary node and update that copy.

When failures of this type occur, typical SAN-based systems are left with two alternatives: First, the first processor may be shut down, forcing the second processor to handle all load, but permitting maintenance of the mirrored data. This is undesirable because there may be significant loss of throughput with the first processor off-line. Second, the first storage node may be shut down, permitting the processors to share the load, but causing the mirrored

datasets to lose synchronization. This is undesirable because synchronization of the datasets is required before the first storage node can be brought back on-line, and because there is a risk of data loss should the second storage node fail before synchronization is completed.

## SOLUTION TO THE PROBLEM

A modified NL_Port (M_Port) has capability to automatically maintain a mirrored dataset on a pair of storage nodes. A second M_Port can perform a write operation to a copy of a mirrored dataset on behalf of a first M_Port should the second M_Port be able to communicate with the first M_port, and the first M_Port be unable to reach that copy of the mirrored dataset. This write by the second M_Port in behalf of the first M_Port is known herein as a surrogate write.

In a first embodiment, surrogate writes are performed by port hardware, without need to involve a node processor in the surrogate write.

In another embodiment of the invention, surrogate writes are performed by a SAN node, thereby enabling surrogate writes when surrogate write requests are received on a port other than that in communication with the target of the surrogate writes.

The invention is applicable to Storage Area Networks (SANs) in general, and is of particular utility for Web-page serving and transaction processing systems.

The foregoing and other features, utilities and advantages of the invention will be apparent from the following more particular description of a preferred

embodiment of the invention as illustrated in the accompanying drawings.

## BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a diagram of a network that may benefit from the present invention;

Figure 2, a flowchart of surrogate writes to a non-mirrored dataset;

Figure 3, a flowchart of surrogate writes maintaining a mirrored dataset;

Figure 4, a block diagram of a port capable of transmitting frames into and receiving frames from a storage area network; and

Figure 5, a flowchart of how the invention handles data write frames encapsulated for transmission to an M_port for surrogate write operation.

## DETAILED DESCRIPTION

A storage area network has a first compute node 100 that has a link 102 to a first switch or hub 104. The first switch or hub 104 also has a link 106 to a first storage node 108, and a link 110 to a second storage node 112. A path therefore exists from the first compute node 100 through the first switch 104 to each of the first storage node 108 and second storage node 112.

Similarly, the network has a second compute node 120 that has a link 122 to the first switch or hub 104 and a link 124 to a second switch or hub 126. The second switch or hub 126 also has a link 128 to the first storage node 108, and a link 130 to the second storage node 112. A path therefore exists from the

second compute node 120 through the second switch 126 to each of the first storage node 108 and second storage node 112.

A dataset is mirrored, such that a first copy 131 of the dataset is maintained on the first storage node 108, and a second copy 132 maintained on the second storage node 112. This dataset is in use by the first compute node 100, and may also be in use by other nodes of the SAN.

At least one path exists through the network for communication between the first compute node 100 and the second compute node 120. In this example network, a path exists from first compute node 100, link 102, switch 104, link 172 to second compute node 120.

The particular network configuration of Figure 1 is by way of example to illustrate the utility and operation of the invention and not by way of limitation. Many other network configurations are possible that may benefit from the invention. Some network configurations that may benefit from the invention may themselves result from failure or overload of network components.

When compute node 100 reads from the dataset, it may read from either the first dataset copy 131 or the second dataset copy 132. When compute node 100 writes to the dataset, it must write to both the first dataset copy 131 and the second dataset copy 132 if synchronization of the datasets is to be maintained.

Consider failure of link 106 between the first switch or hub 104 and the first storage node 108.

In this event, the path from first compute node 100 through switch 104 to first storage node 108 and the first dataset copy 131 will also fail. Since the path from first compute node 100 through switch 104 to

second storage node 112 and the second dataset copy 132 is still operational, first compute node 100 can continue to read the dataset by reading the second copy 132. Since the path to the first dataset copy 131 has failed, compute node 100 can not ordinarily write to first dataset copy 131, which may result in loss of synchronization of the dataset copies.

In many SANs, the compute nodes, such as first compute node 100 and second compute node 120, are in communication with each other. In the example of Figure 1, first compute node 100 may communicate with second compute node 120 through first switch 104 by way of links 102 and 122. In the example network configuration, first compute node 100 may also communicate with second compute node 120 through network hardware separate from the SAN, such as an ethernet or other local area network 136.

With only link 106 failed, second compute node 120 still has a path through links 124 and 128, and switch 126, to the first storage node 108 and the first dataset copy 131.

In a network embodying the present invention, when first compute node 100 can not reach first storage node 108, second compute node 120 can reach first storage node 108, and first compute node 100 can reach second compute node 120; the second compute node 120 performs surrogate write operations in behalf of first compute node 100. This permits maintenance of synchronization between the first copy 131 and the second copy 132 of the dataset.

Surrogate read or write operations may also be performed to non-mirrored datasets, provided that a path exists from the compute node desiring the read or

write to a compute node having a path to the destination device.

Each compute node maintains a list of paths to storage nodes. This list includes status of the paths. It is known that path status can change to failed should a problem occur with a link, switch, or other network device.

When surrogate writes are enabled and a compute node desires to write a dataset 200 (Figure 2), that node checks 202 the path status to the storage node on which the dataset is stored. If that path has a status of "path OK" 204, a write is attempted 206 to the dataset on that node. If the write succeeds, all is well. If the write fails 208 for reasons that are likely to be a result of a failed path to the storage node, such as a fibre channel timeout error, the node looks for a path 210 to a second compute node, and verifies that that path has a status of "path ok". If that path has status indicating it is failed, the node looks 212 and 214 for any other compute nodes to which it might have a good path. If no such path is found, the write is declared 215 to have failed.

Once a compute node is identified to which there is a good path, a query is sent 216 to that compute node asking if it has a valid path to the storage node on which the dataset is stored. If that query fails 218 for reasons that are likely to be a result of a failed path to the node, such as a fibre channel timeout error, the node looks 212 and 214 for any other compute nodes to which it might have a good path.

If the second compute node reports that it has no "OK" path 220 to the target storage node, the node looks 212 and 214 for other compute nodes that might

have a path to the target storage node. If the second compute node reports that it has an "OK" path to that target node, the node encapsulates 222 a write request into suitable frames and transmits those frames to the second compute node. The second compute node then relays that write request to the target node and relays any reply back to the compute node desirous of the write.

If the write occurs correctly 224, the compute node continues to process data. If the write is reported as having failed or timed out, the write is declared failed 215 and suitable error routines invoked.

Writes to a mirrored data set are handled similarly. When a write request occurs 300, the source node checks its path status 302 to both storage nodes having copies of the dataset. If both paths have good status 304, writes occur in the normal manner 306. If either write fails 308 for reasons, such as timeout, that could be related to a bad path, a check 310 is made to determine if both failed or if only one failed. If both write attempts failed, a both-paths failed routine is invoked (not shown).

If, when the path status was checked 302 to both storage nodes, one path was broken and the other was OK, a write is generated 312 to the storage node that can be reached. If that write fails for timeout or other reasons that could be related to a bad path, the both-paths failed routine is invoked (not shown). If that write succeeds, the source node checks 314 for an OK path to a compute node. If the compute node first tried has no valid path, the source node searches 316 and 318 for a compute node to which it has a valid path. If no compute node to which there is a valid

path can be found, the mirror set is declared broken 320; such that when paths are restored an attempt will be made to re-synchronize the mirror set.

Once a valid path is found to a compute node, a query is made 322 of that compute node to determine if it has a valid path to the target storage node and to determine if that node supports surrogate writes. If that query results in a reply indicating that surrogate writes are not supported or that there is no valid path 326, or the query times out or fails for other reasons indicative of a failed path 324, the source node may continue to search 316 and 318 for another compute node that has a valid path and supports surrogate writes.

If the compute node reports that it has a valid path and supports surrogate writes, the source node encapsulates a write request into suitable frames, and transmits 328 those frames to the compute node. That node then relays the write request to the target node, and relays any reply from the target node to the source node.

Any reply from the target node is inspected to determine 330 if the write succeeded. If the write was not successful, the mirror set is reported broken 320.

It is anticipated that the present invention can be implemented in driver software of a compute node, or alternatively can be implemented in firmware of an HBA, such as a dual-port HBA.

A dual port Host Bus Adapter (HBA) (Figure 4) typically has a port processor 400, a memory system 402 for storing frames, a DMA (Direct Memory Access) transfer system 404 and other hardware for communicating with its host (not shown), and first 406

and second 408 serializer and deserializer hardware. Each set of serializer and deserializer hardware is coupled to a network transmitter 410 and 412, and to a network receiver 414 and 416.

A dual-port HBA like that illustrated in Figure 4 implements the connection of second compute node 120 (Figure 1) to links 122 and 124. A second, similar, HBA implements the connection of first compute node 100 to link 102. Each HBA is capable of maintaining a mirror set under control of firmware located in its memory system 402 and running on its port processor 400, and of implementing the method of requesting surrogate writes previously described.

Whenever the dual-port HBA receives frames 500 (Figure 5), the frames are inspected 502 to determine the frame type. If they are path query frames 504 from a source node, as sent in steps 322 (Figure 3) or 216 (Figure 2) as previously described, the status of any path to the target node is determined 506, and checked 508. If a valid path exists, a reply frame is constructed 510 by the port processor 400 indicating that surrogate writes are supported and that the path is OK, otherwise a frame is constructed 512 indicating that the path does not exist or is not OK. This constructed frame is sent 514 to the source node.

If the frame was not a path query, the frame is checked 520 to see if it encapsulates a write request. If it does, the write request is de-encapsulated and forwarded 522 to the target node. If the frame does not encapsulate a write request, it is checked 526 to see if it is a response to a forwarded write request. If it is such a response, the write status is relayed 528 to the source node.

While the invention has been particularly shown and described with reference to a preferred embodiment thereof, it will be understood by those skilled in the art that various other changes in the form and details may be made without departing from the spirit and scope of the invention.

## CLAIMS

What is claimed is:

1. A host bus adapter for interconnecting a computer system to a storage area network comprising

hardware for transmission of frames;

hardware for reception of frames;

memory for storage of frames;

a processor for processing frames, the processor coupled to the hardware for transmission of frames, the hardware for reception of frames, and the memory for storage of frames;

wherein the processor is capable of inspecting frames for encapsulated write requests and, if encapsulated write request frames are found, de-encapsulating the write request and forwarding the write request through the hardware for transmission of frames to a target node of the write request.

2. The host bus adapter of Claim 1, wherein the processor further inspects frames for responses to encapsulated write requests and, when such responses are found, forwards the responses to a source node from which the original encapsulated write request frames came.

3. The host bus adapter of Claim 2, wherein the host bus adapter is capable of simultaneous connection to at least two links, having a transmitter and a receiver for coupling to each link.

4. The host bus adapter of Claim 2, wherein the host bus adapter is capable of responding to a query frame, the query frame containing a request for status of any

path that might exist from the host bus adapter to a specified target node.

5. The host bus adapter of Claim 1, wherein the host bus adapter is capable of maintaining a mirrored dataset on at least two target nodes.

6. The host bus adapter of Claim 5, wherein the host bus adapter is capable of determining that it has no direct path to a target node of the at least two target nodes, and, when no direct path exists, is capable of requesting that another node perform a surrogate write to the target node.

7. The host bus adapter of Claim 5, wherein the host bus adapter is capable of scanning nodes to determine a node capable of performing a surrogate write to the target node.

8. A node for connection to a storage area network comprising

hardware for transmission of frames;

hardware for reception of frames;

memory;

at least one processor for processing frames, the processor coupled to the hardware for transmission of frames, the hardware for reception of frames, and the memory for storage of frames;

wherein the processor is capable of inspecting frames for encapsulated write requests and, if encapsulated write request frames are found, de-encapsulating the write request and forwarding the write request through the hardware for transmission of frames to a target node of the write request.

9. The node of Claim 8, wherein the processor further inspects frames for responses to encapsulated write requests and, when such responses are found, forwards the responses to a source node from which the original encapsulated write request frames came.

10. The node of Claim 9, wherein the node is capable of simultaneous connection to at least two links, having a transmitter and a receiver for coupling to each link.

11. The node of Claim 9, wherein the node is capable of responding to a query frame, the query frame containing a request for status of any path existing from the node to a specified target node.

12. The node of Claim 8, wherein the node is capable of maintaining a mirrored dataset on at least two target nodes.

13. The node of Claim 12, wherein the node is capable of determining that it has no direct path to a target node of the at least two target nodes, and, when no direct path exists, is capable of requesting that another node perform a surrogate write to the target node.

14. The node of Claim 13, wherein the node is capable of scanning nodes of a network to determine a node capable of performing a surrogate write to the target node.

15. A computer network comprising:

a first node;

a second node;

a first target node;

network interconnect providing communication between the first node and the second node, and providing communication between the second node and the first target node;

wherein the second node is capable of inspecting incoming frames for encapsulated write requests and, if encapsulated write request frames are found, de-encapsulating a write request from the encapsulated write request frames and forwarding the write request to the first target; and

wherein the second node further is capable of inspecting frames received from the first target node for responses to previously forwarded encapsulated write requests and, when responses to previously forwarded encapsulated write requests are found, forwarding the responses to the first node.

16. The computer network of Claim 15, wherein the second node is capable of responding to a path query frame with a status of a path from the second node to the first target node.

17. The computer network of Claim 16, wherein the network interconnect is fibre channel compatible.

18. The computer network of Claim 16, further comprising a second target node and the network interconnect further provides communication between the second target node and the first node; and

wherein the first node is capable of maintaining a mirrored data set having a copy on the first target node and the second target node.

19. A method of performing writes by a first node of a storage area network to a mirrored dataset, the

dataset comprising a first copy on a first storage node of the storage area network and a second copy on a second node of the storage area network, the storage area network having a surrogate-capable node, the method comprising the steps of:

checking a status of a first path from the node to the first storage node and of a second path from the node to the second storage node;

the first path has good status and the second path has bad status, then:

performing a write to the first copy of the mirrored dataset over the first path;

polling the surrogate-capable node to determine whether the surrogate-capable node has a path having good status to the second storage node;

if the surrogate-capable node has a path having good status to the second storage node, encapsulating a write request to the second copy and transmitting that encapsulated write request to the surrogate-capable node; and

de-encapsulating the encapsulated write request to the second copy and forwarding it from the surrogate-capable node to the second storage node.

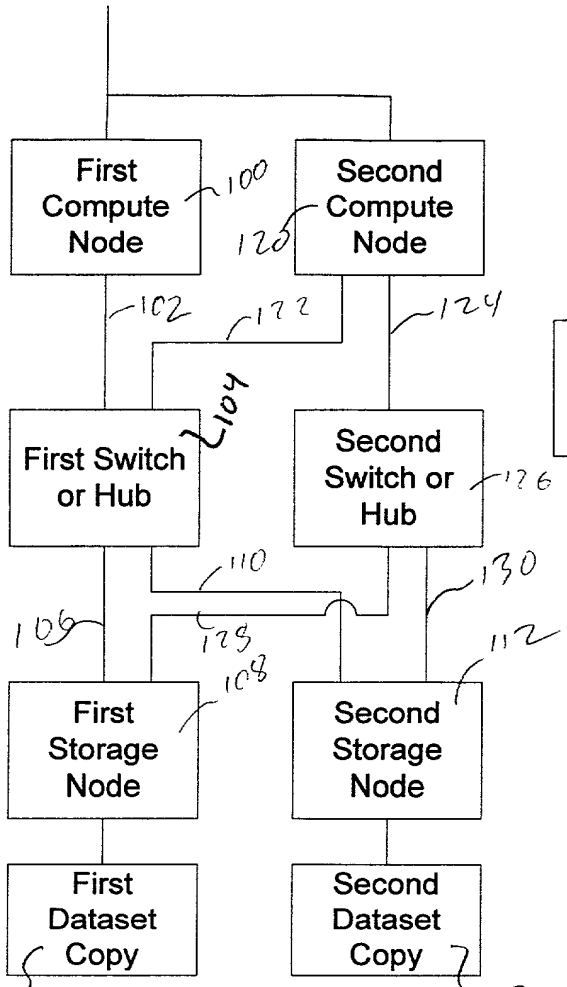
20. The method of Claim 19, further comprising the steps of

transmitting a response to the write request to the second copy from the second storage node to the surrogate-capable node, and of

forwarding the response to the write request from the surrogate-capable node to the first node.

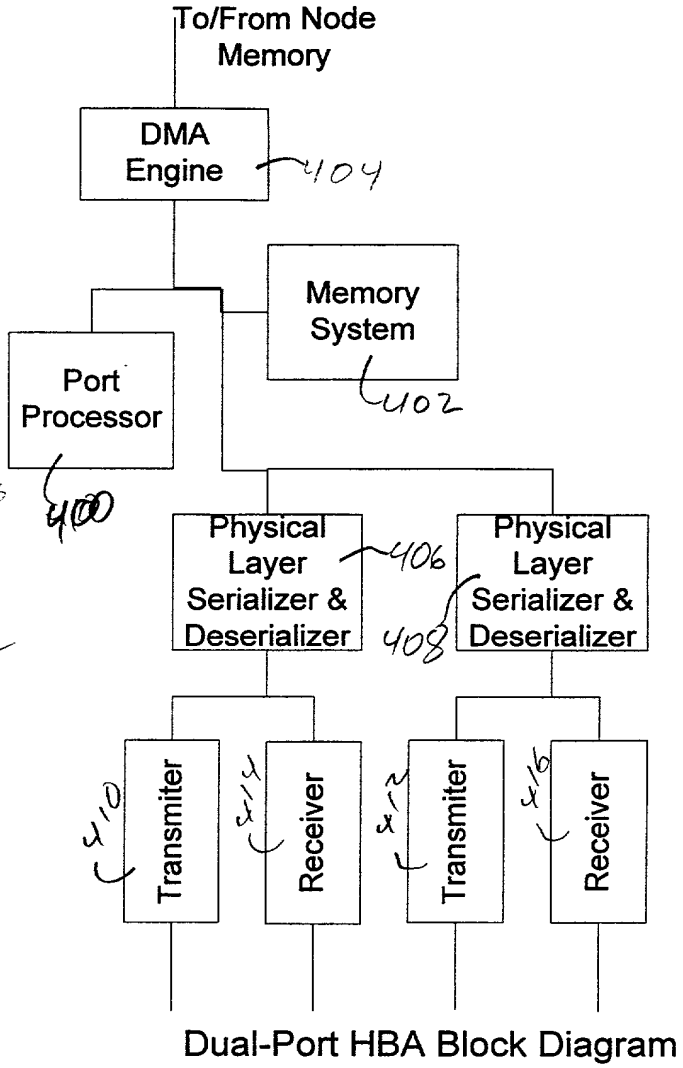
## ABSTRACT

A host bus adapter for interconnecting a computer system to a storage area network has an embedded processor for processing frames. When frames are received by the processor, it inspects frames for encapsulated write requests and, if encapsulated write request frames are found, de-encapsulates the write request and forwards the write request to a target node of the write request.




Example SAN Configuration

Figure 1




Dual-Port HBA Block Diagram

Figure 4

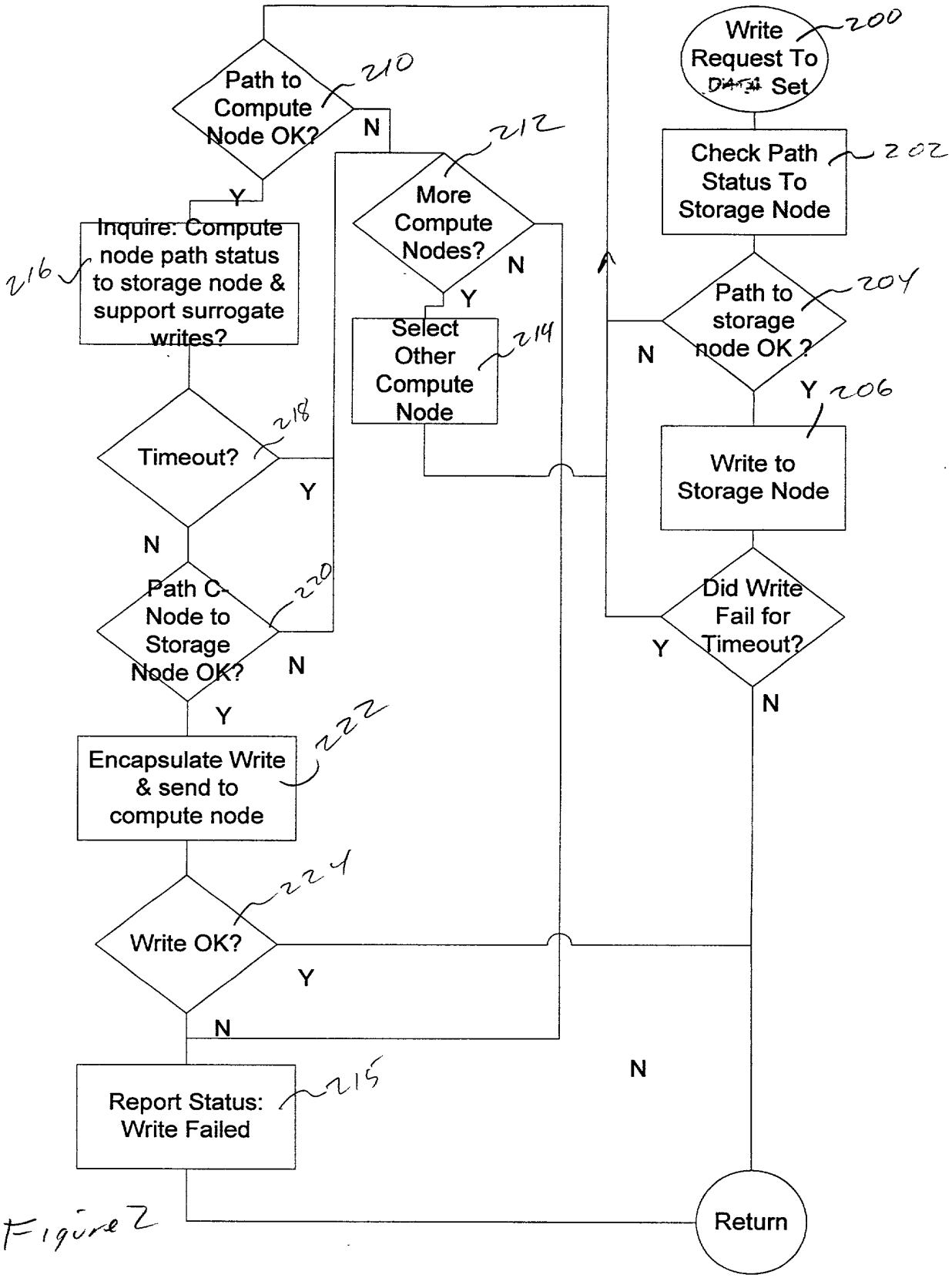

Write Request To Data Set
200
Check Path Status To Storage Node
202
Path to storage node OK ?
204
N
Y
206
Write to Storage Node
Did Write Fail for Timeout?
Y
N
Path to Compute Node OK?
210
N
Y
More Compute Nodes?
212
N
Y
Select Other Compute Node
214
Inquire: Compute node path status to storage node & support surrogate writes?
216
Timeout?
218
Y
N
Path C-Node to Storage Node OK?
220
N
Y
Encapsulate Write & send to compute node
222
Write OK?
224
Y
N
Report Status: Write Failed
215
Return

Figure 2

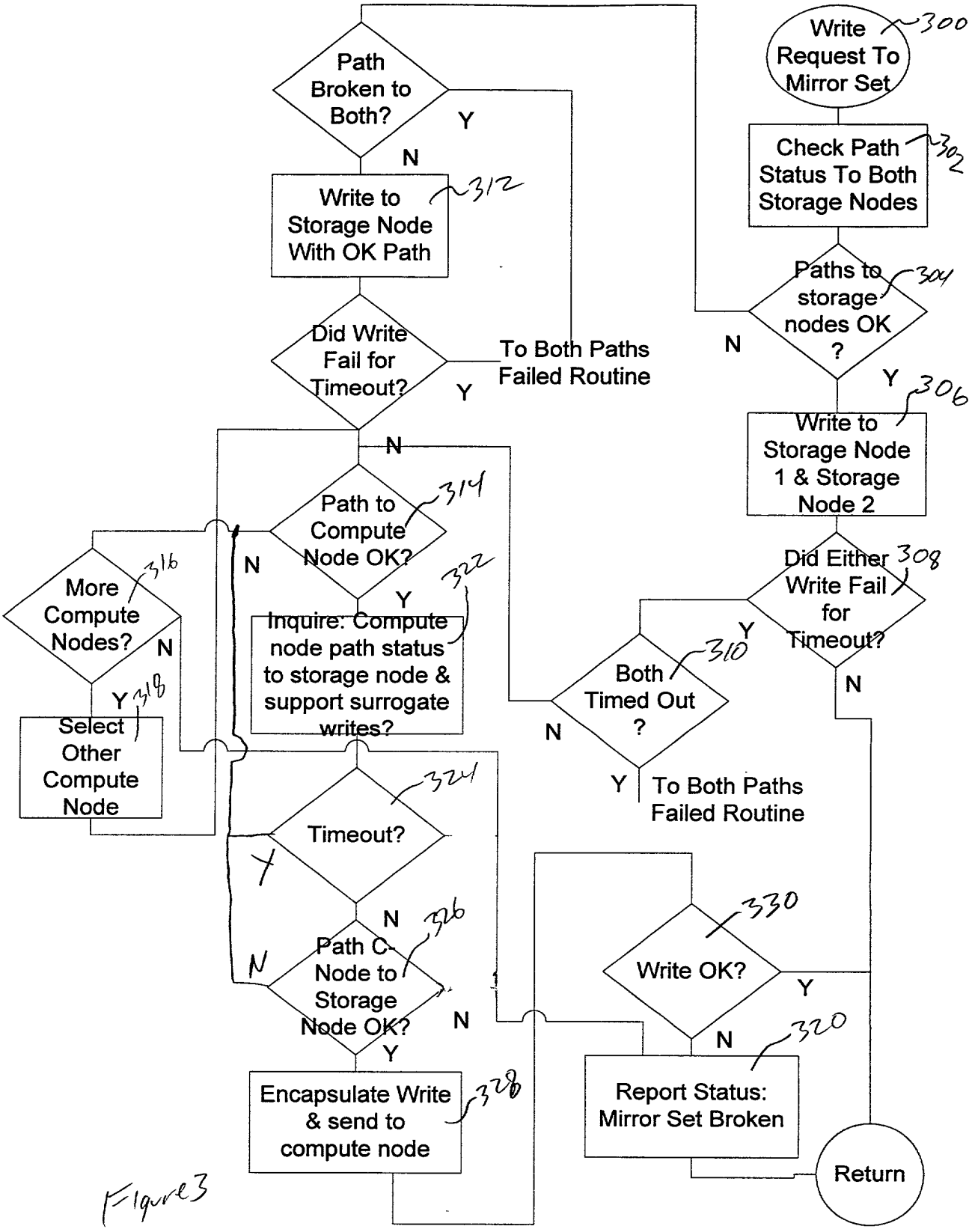

Write Request To Mirror Set
300
Check Path Status To Both Storage Nodes
302
Paths to storage nodes OK ?
304
N
Y
Write to Storage Node 1 & Storage Node 2
306
Did Either Write Fail for Timeout?
308
Y
N
Both Timed Out ?
310
N
Y
To Both Paths Failed Routine
Path Broken to Both?
Y
N
Write to Storage Node With OK Path
312
Did Write Fail for Timeout?
To Both Paths Failed Routine
Y
N
Path to Compute Node OK?
314
N
Y
More Compute Nodes?
316
N
Y
Select Other Compute Node
318
Inquire: Compute node path status to storage node & support surrogate writes?
322
Timeout?
324
Y
N
Path C-Node to Storage Node OK?
326
N
Y
Encapsulate Write & send to compute node
328
Write OK?
330
Y
N
Report Status: Mirror Set Broken
320
Return

Figure3

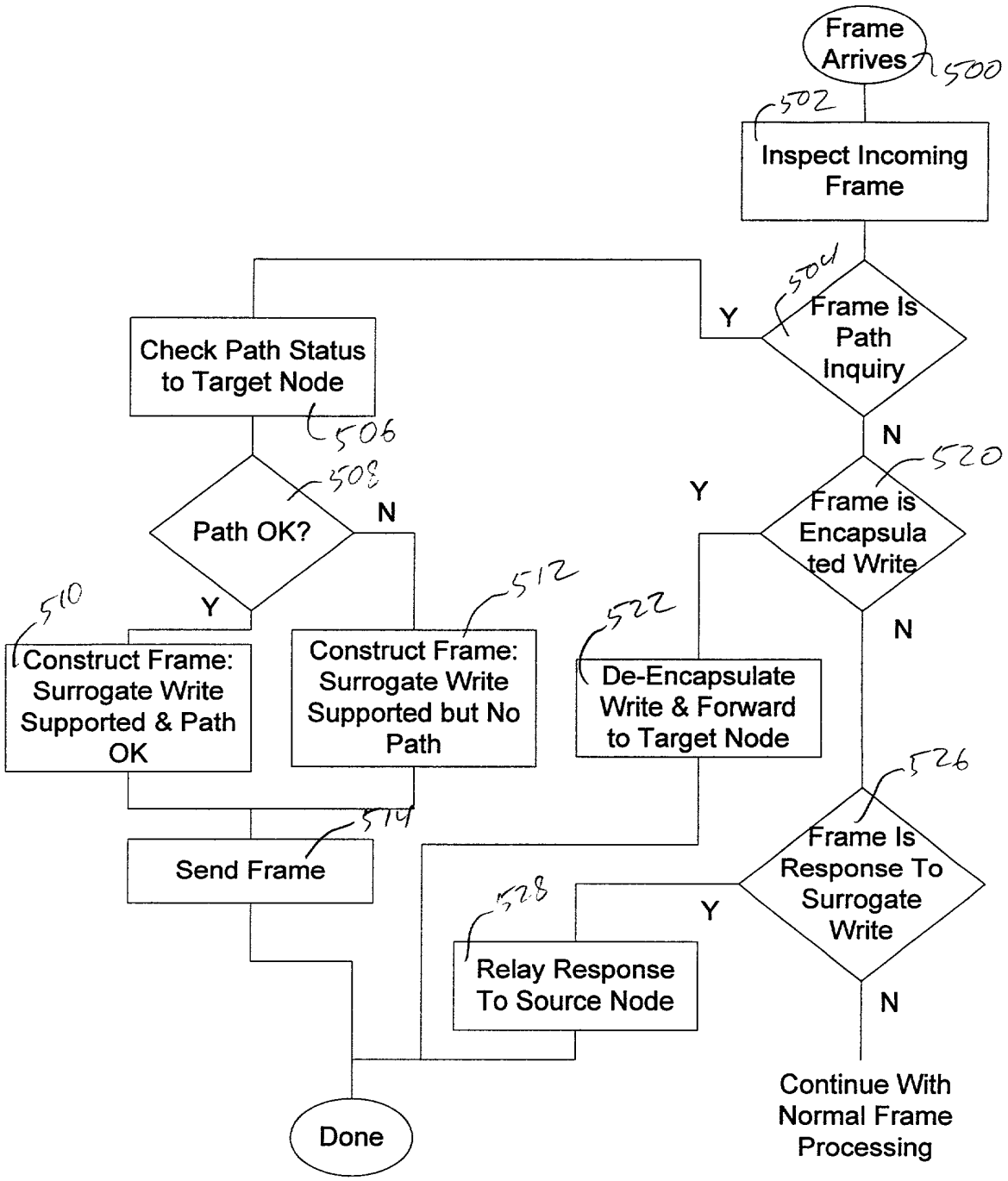

Frame Arrives
500
502
Inspect Incoming Frame
504
Frame Is Path Inquiry
Y
N
Check Path Status to Target Node
506
508
Path OK?
N
Y
510
Construct Frame: Surrogate Write Supported & Path OK
512
Construct Frame: Surrogate Write Supported but No Path
514
Send Frame
520
Frame is Encapsulated Write
Y
N
522
De-Encapsulate Write & Forward to Target Node
526
Frame Is Response To Surrogate Write
Y
N
528
Relay Response To Source Node
Done
Continue With Normal Frame Processing

Figure 5

PATENT
Attorney Docket No. P00-3251
Client/Matter No. 68854.0152
Express Mail No. EL700672270US
Joint Inventors
Original

# DECLARATION

As a below named inventor, I hereby declare that: my residence, post office address, and citizenship are as stated below next to my name. I believe I am the original, first, and sole inventor (if only one name is listed below) or a joint inventor (if plural inventors are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled:

SYSTEM, MACHINE, AND METHOD FOR MAINTENANCE OF MIRRORED DATASETS THROUGH SURROGATE WRITES DURING STORAGE-AREA NETWORK PARTIAL CONNECTIVITY EVENTS

as described in the specification ☒ attached or ☐ of patent Application Serial No. ----------
filed _____ and amended on _____.

I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claims, as amended by any amendment referred to above; that I do not know and do not believe the same was ever known or used in the United States of America before my or our invention thereof, or patented or described in any printed publication in any country before my or our invention thereof or more than one year prior to this application; that the invention has not been patented or made the subject of an inventor's certificate issued before the date of this application in any country foreign to the United States of America on an application filed by me or my legal representative or assigns more than twelve months prior to this application; and that I acknowledge the duty to disclose information of which I am aware which is material to the examination of this application in accordance with Title 37, Code of Federal Regulations ' 1.56(a). Such information is material when it is not cumulative to information already of record or being made of record in the application, and

(1) it establishes, by itself or in combination with other information, a prima facie case of unpatentability of a claim; or
(2) it refutes, or is inconsistent with, a position the applicant has taken or may take in:
(i) opposing an argument of unpatentability relied on by the Office, or
(ii) asserting an argument of unpatentability.

I hereby claim foreign priority benefits under Title 35, United States Code ' 119 of any foreign application(s) for patent or inventor's certificates listed below and have also identified below any foreign application(s) having a filing date before that of the applications(s) on which priority is claimed:

| COUNTRY | APPLICATION NUMBER | Date Filed | Priority Claimed under 35 USC 119 |
|---|---|---|---|
| | | | ☐ YES ☐ NO |

I hereby claim the benefit under Title 35 United States Code ' 120 of any United States application(s) listed below and, insofar as any subject matter of any claim of this application is not disclosed in the prior United States Application, I acknowledge the duty to disclose material information as defined in Title 37, Code of Federal Regulations ' 1.56(a) which occurred between the filing date of the prior application and the national PCT international filing date of this application.

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

| FULL NAME OF SOLE OR FIRST INVENTOR | INVENTOR'S SIGNATURE | DATE |
|---|---|---|
| Greg Pellegrino | Greg Pellegrino | 11/01/00 |
| RESIDENCE<br>19227 Cougar Peak Drive, Tomball, TX 77375 | | CITIZENSHIP<br>USA |
| POST OFFICE ADDRESS<br>Same | | |
| FULL NAME OF SECOND JOINT INVENTOR | INVENTOR'S SIGNATURE | DATE |
| Thomas Grieff | Thomas Grieff | 11/01/00 |
| RESIDENCE<br>11100 Louetta Road #324, Houston, TX 77070 | | CITIZENSHIP<br>USA |
| POST OFFICE ADDRESS<br>Same | | |

Express Mail No.: EL700672270US

# IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

*Applicant/Patentee:*

Greg Pellegrino and Thomas Grieff

*Serial No.:* ----------

*Date Filed:* Herewith

*For:* SYSTEM, MACHINE, AND METHOD FOR MAINTENANCE OF MIRRORED DATASETS THROUGH SURROGATE WRITES DURING STORAGE-AREA NETWORK PARTIAL CONNECTIVITY EVENTS

*Attorney File No.:* 68854.0152

*Digital Docket No.:* P00-3251

## POWER OF ATTORNEY BY ASSIGNEE

Under the provisions of 37 C.F.R. § 3.71, the undersigned assignee of record of the entire interest in the above-identified patent/patent application by virtue of an assignment recorded (check as applicable):

☒ Concurrently Herewith
☐ Date Recorded _____
☐ Reel _____ Frame _____

elects to conduct the prosecution of the application/maintenance of the patent to the exclusion of the inventor(s). The undersigned hereby declares that she has reviewed the above-referenced assignment and hereby declares that, to the best of her knowledge, title is in the Assignee, and further declares that all statements made herein of her own knowledge are true and that all statements made on information and belief are believed to be true. The assignee hereby revokes any previous powers of attorney and appoints the following to prosecute this application/maintain this patent and transact all business in the Patent and Trademark Office connected therewith:

*(Prosecuting Attorney List)*

| | |
|---|---|
| Irene Kosturakis, Reg. No. 33,724 | Stuart T. Langley, Reg. No. 33,940 |
| Rich Lange, Reg. No. 27,296 | Carol W. Burton, Reg. No. 35,465 |
| Louis Brucculeri, Reg. No. 38,834 | Steven Kent Barton, Reg. No. 36,445 |
| Sarah T. Harris, Reg. No. 35,891 | E. Michael Byorick, Reg. No. 34,131 |
| Joseph Arrambide, Reg. No. 39,589 | Matthew G. Dyor, Reg. No. 42,278 |
| Keith Lutsch, Reg. No. 31,851 | Steven C. Petersen, Reg. No. 36,238 |
| Theodore S. Park, Reg. No. 26,971 | Sarah S. O'Rourke, Reg. No. 41,226 |
| William J. Kubida, Reg. No. 29,664 | Kent A. Lembke, Reg. No. 44,866 |

Please direct all communications relative to this application to the following addressee:


WILLIAM J. KUBIDA
Hogan & Hartson LLP
One Tabor Center
1200 17th Street, Suite 1500
Denver, Colorado 80202
(719) 448-5909


**ASSIGNEE**
COMPAQ COMPUTER CORPORATION

Date: 01 Nov 2000

BY: Diane H. Strong
NAME: Diane H. Strong
TITLE: Administrator, Patents
Authorized To Sign This Document On Behalf Of Compaq Computer Corp.